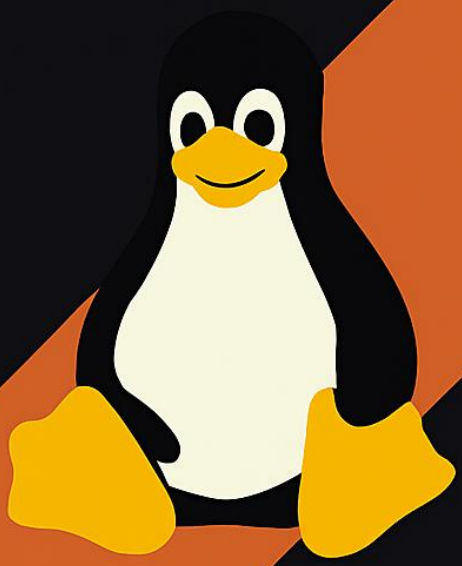
Linux Essentials Cheat Sheet



Linux:

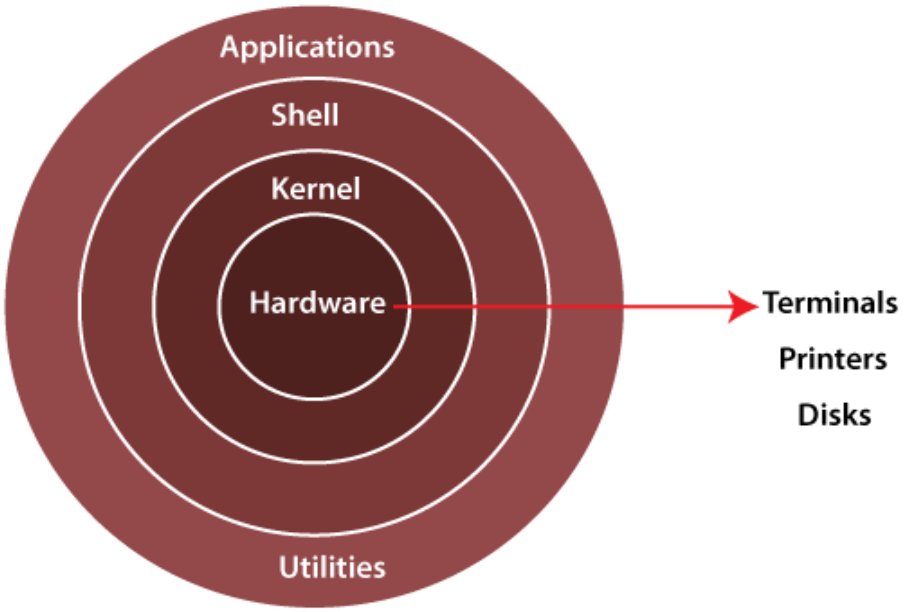
Linux is an open-source, Unix-like operating system kernel. It manages hardware resources, runs applications, and provides both CLI (Command-Line Interface) and GUI (Graphical User Interface) through shells and desktop environments.

Why Linux:

- Powers servers, DevOps, cloud, embedded systems, super computers.
- Secure, stable, & widely used in IT

Key components:

1. kernel: core OS that manages CPU, memory, devices
2. shell: interface for commands (bash, zsh)
3. file system: Hierarchical structure (/home, /etc, /var etc.,)
4. Utilities: programs to manage files, Processes, networks



🗁Linux file system:

/

├── boot → Bootloader files (kernel, initrd, grub)

├── bin → Essential user commands (ls, cp, mv, cat)

├── sbin → System binaries (fsck, reboot, ifconfig)

├── etc → Configuration files (/etc/passwd, /etc/ssh/sshd_config)

├── home → User home directories (/home/user)

├── root → Root user's home directory

├── lib → Shared libraries required by /bin and /sbin

├── usr → User programs, libraries, docs (/usr/bin, /usr/lib)

├── var → Variable data (logs, mail, spool, cache)

├── tmp → Temporary files

├── opt → Optional / third-party software

├── mnt → Temporary mount point

├── media → Mount point for removable media (USB, CD-ROM)

├── dev → Device files (disks, terminals)

└── proc → Virtual filesystem (process and kernel info)

🏔Paths:

- Absolute Path: /home/user/file.txt
- Relative Path: ./file.txt

💻Basic commands & file operations:

1. pwd → Print current directory
2. ls → list files & directories in pwd
3. ls -R → list files in sub-directories as well
4. ls -a → shows hidden files.
5. ls -al → long lists files & directories with detailed info via permissions, size owner etc;
6. ls -lt → time sequence
7. ls -alt → list all files including hidden ones, sorted by time
8. cd directoryname → changes to directory.
9. cd .. → move one level up

7 columns

1 → File type + permissions

2 → Number of hard links

3 → Owner (user)

4 → Group

5 → File size (in bytes, not KB)

6 → Last modified date & time

7 → File name

10. cat > filename → creates a new file, cat >> filename → add data
11. cat filename → displays the file content
12. cat file1 file2 > file3 → joins 2 files & store o/p in a new file (file3)

13. touch filename → creates a file
14. rm filename → deletes a file
15. cp source destination → copies file from source to destination path
16. mv source destination → moves file from source destination path
17. find / -name filename → finds a file by its name
18. file filename → determines file type
19. less filename → view the file content page by Page
20. head filename → view the first ten lines of a files
21. tail filename → views last ten lines of a file
22. lsof → shows which files are opened by which process
23. du -h --max-depth=1 → shows the size of each dir. Use --max-depth = 1 to limit the o/p to the current dir & its immediate children.
24. fdisk → disk partition manipulation command.

filter commands: more, less, head, tail, sort, sed, cut

sort filename && sort -n filename → numeric order

sort -u filename → eliminate duplicate


We can edit a file by 3 ways: vi/vim/nano

In vi we have 3 types of modes: command/ insert/ extended command

Command mode:

- G → end of the file
- gg → start of the file
- w → line by line forward
- b → line by line backward
- u → undo the single line
- U → undo changes made to the current line since you entered it
- p → paste
- P → paste
- yy → one line copy
- nyy → n lines copy
- dw → line by line delete
- x → letter by letter delete
- ndd → to delete the line.

Insert mode:

- i → insert at cursor position
- I → insert at beginning of the line
- a → insert at next letter
- A → Insert at end of the line
- o → open a new line below cursor
- O → open a new line above cursor

Extended Command mode:

- wq → save & quit
- x → save & quit
- w → save only
- w! → save forcefully
- q → quit

- q! → quit forcefully
- wq! Save & quit forcefully
- :set nu → set numbers to the lines
- :set nonu → set no numbers to the lines
- :n → jump to line number n

🗁 Directory operations:

1. mkdir directoryname → creates a new dir in pwd
2. rmdir directoryname → deletes a directory
3. cp -r source destination → copies directories recursively
4. mv olddir newdir → rename directories
5. find / -type d -name "directoryname"→ finds a directory
6. rm -rf, rm -f → removes dir with files starting from root

🔐 File permissions:

Types of permission r=read w=write x=execute

check permissions by ls -l

1. change permissions → chmod 775 script.sh && chmod u+x file.sh
2. change ownership → chown user:group file.txt

r=4, w=2, x=1 765, 400 → read only

3. chmod octal filename → change permissions of octal which can be between 0 to 7
4. chown ownername filename → change owner
5. chgrp groupname filename → change group owner.

User management

adduser is more user-friendly; useradd is lower-level and scriptable

1. whoami → show current user
2. id → show UID, groups
3. adduser user1 → add new user
4. passwd user1 → change password
5. su user1 → switch to user1 account.
6. sudo cmd → run command as root
7. useradd <username> → creates a user
8. usermod → modify a user
9. userdel → delete a user, userdel -r username
10. passwd → assign password for user. passwd <username>
11. su - → switch to root (if password provided).

Group management:

1. groupadd → create a group
2. gpasswd → assign password
3. gpasswd -r <groupname> → removes password
4. groupmod -n oldgroupname newgroupname → changing group name

/etc/passwd → it contains local user details

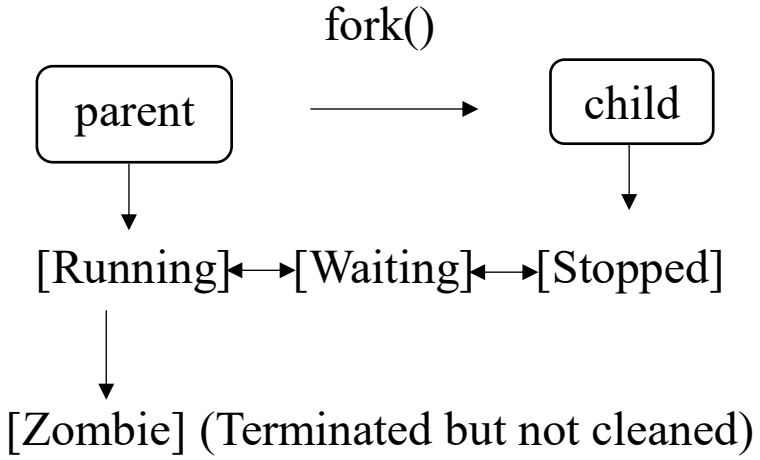
/etc/group → it contains local group properties

/etc/gshadow → it contains local group password properties

/etc/shadow → it contains local user password properties

⚙ Process operations:

1. ps → display your currently active processes
2. top → display all running processes (live system monitor)
3. kill PID → kills process with given pid
4. pkill name → kills the process with the given name
5. bg → resumes suspended jobs without bringing them to foreground
6. fg → brings the most recent job to foreground
7. fg n → brings job n to the foreground
8. renice +n [pid] → change the priority of a running process.
9. &>filename → redirects both the stdout and the stderr to the file filename.
10. 1>filename → redirect the stdout to file filename.
11. 2>filename → redirect stderr to file filename.



📦 Package management:

1. sudo apt-get update → updates package lists for upgrades
2. sudo apt-get upgrade → upgrades all upgradable packages

3. sudo apt-get install pkgname → install pkgname
4. sudo apt-get remove pkgname → removes pkgname

🌐 Networking:

1. ping host → ping a host and outputs results
2. whois domain → get whois information for domain
3. dig domain → get DNS information for domain
4. netstat -pnltu → display various network related information such as network connections, routing tables, interface statistics etc.
5. ifconfig → displays IP addresses of all network interfaces
6. ssh user@host → remote login into the host as user
7. scp source user@host:/path → transfers files between hosts over ssh
8. wget url → download files from the web
9. curl url → sends a request to a URL and returns the response
10. traceroute domain → prints the route that a packet takes to reach the domain.
11. mtr domain → mtr combines the functionality of the traceroute and ping programs in a single network diagnostic tool.
12. ip addr (modern alternative to ifconfig).
13. ip route (to view/manage routes)
14. ss → another utility to investigate sockets. It's a more modern alternative to netstat.
15. Nmap → network exploration tool and security scanner.

16. tree → used to see all subdir and files. We have to install tree package: sudo apt install tree

💿Disk Usage

1. df → show disk usage
2. du → show directory space usage
3. free → show memory and swap usage
4. whereis app → show possible locations of app
5. lsblk → show block devices
6. df -h → mounted disks
7. mount /dev/sdb1 /mnt → mount disks
8. umount /mnt → unmount
9. dd if=/dev/zero of=/tmp/output.img bs=8k count=256k → create a file of a certain size for testing disk speed.
10. hdparm -Tt /dev/sda → measure the read speed of your hard drive.

System Info:

1. date → show the current date and time
2. cal → show this month's calendar
3. uptime → show current uptime
4. w → display who is online
5. whoami → who you are logged in as
6. uname -a → show kernel information
7. df -h → disk usage in human readable format
8. du -sh → disk usage of current directory in human readable format

9. free -m → show free and used memory in MB

📝 Text Processing:

1. tr → translate or delete characters (e.g., tr a-z A-Z).
2. uniq → filter out duplicate lines (often used with sort).
3. grep pattern files → search for pattern in files
4. grep -r pattern dir → search recursively for pattern in dir
5. grep -i pattern file → case-insensitive search
6. grep -v pattern file → invert match (exclude)
7. command | grep pattern → pipe the output of command to grep for searching
8. echo 'text': Prints text
9. sed 's/string1/string2/g' filename → replaces string1 with string2 in filename
10. diff file1 file2 → compares two files and shows the differences
11. wc filename → count lines, words, and characters in a file
12. awk → a versatile programming language for working on files.
13. sed -i 's/string1/string2/g' filename → replace string1 with string2 in filename. The -i option edits the file in-place.
14. cut -d':' -f1 /etc/passwd → cut out the first field of each line in /etc/passwd, using colon as a field delimiter.

🗜️ Archives and Compression:

1. tar cf file.tar files → create a tar named file.tar containing files
2. tar xf file.tar → extract the files from file.tar
3. gzip file → compresses file and renames it to file.gz
4. gzip -d file.gz → decompresses file.gz back to file
5. zip -r file.zip files → create a zip archive named file.zip
6. unzip file.zip → extract the contents of a zip file
7. tar -cvf archive.tar /path/to/dir/ → create a tar archive.
8. tar -xvf archive.tar → extract a tar archive.
9. tar -jcvf archive.tar.bz2 dirname/ → create a compressed bz2 archive.
10. tar -jxvf archive.tar.bz2 → extract a bz2 archive.

🔱Environment Variables:

1. unset VAR → remove a variable
2. env → display all environment variables
3. echo $VARIABLE → display the value of an environment variable
4. export VARIABLE=value → set the value of an environment variable
5. alias new_command='old_command options' → create a new command that executes the old command with the specified options.
6. echo $PATH → print the PATH environment variable.
7. export PATH=$PATH:/new/path → add /new/path to the PATH.

📋Logs:

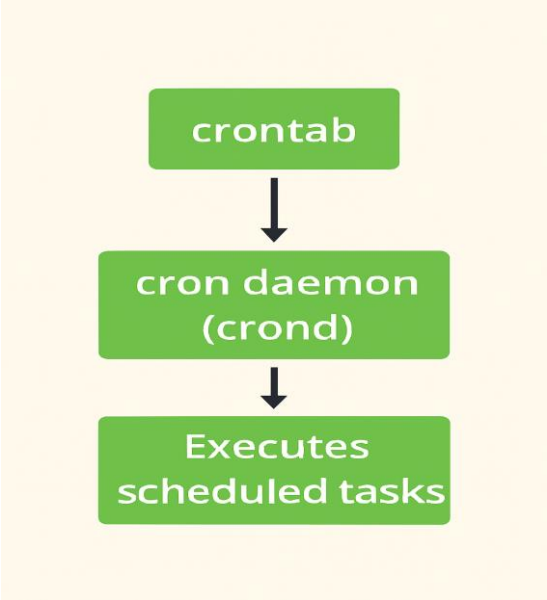
Location /var/log/

Important logs:

1. /var/log/syslog (system messages).
2. /var/log/dmesg (kernel ring buffer).
3. System logs → /var/log/auth.log
4. Webserver logs → /var/log/nginx/
5. journalctl -u <service>, grep “error” logfile.log

⏰Job Scheduling (Cron Jobs):

- Cron = A daemon (background process) that runs scheduled tasks.
- Cron job = A command or script that runs automatically at a specific time/date/interval.



Example uses:

- Take daily backups
- Run monitoring scripts
- Clean logs every week
- Schedule system updates

Cron Syntax: A cron job has 5-time fields + command

```
* * * * * command-to-run
│ │ │ │ │
│ │ │ │ └── Day of week (0-6, Sunday=0)
│ │ │ └──── Month (1-12)
│ │ └────── Day of month (1-31)
│ └──────── Hour (0-23)
└────────── Minute (0-59)
```

1. crontab -l → list all your cron jobs
2. crontab -e → edit your cron jobs
3. crontab -r → remove all your cron jobs
4. crontab -v → display the last time you edited your cron jobs
5. crontab filename → install a cron job from a file
6. @reboot command → schedule a job to run at startup

🚀Services & Systemd:

1. systemctl start nginx → start service
2. systemctl stop nginx → stop service
3. systemctl enable nginx → auto start on boot
4. systemctl status nginx → check status
5. systemctl restart nginx → restart service.
6. systemctl disable nginx → prevent auto-start on boot
7. systemctl is-enabled nginx → check if enabled

8. journalctl -xe → view recent logs with errors.

🛡️Linux Security:

Firewall (UFW in Ubuntu):

1. sudo ufw enable
2. sudo ufw allow 22/tcp
3. sudo ufw status
4. sudo ufw deny 23 → block a port
5. sudo ufw delete allow 22/tcp → remove a rule

🔗Linux File System Advanced:

1. Hard Link: Another name for the same file (points to the same inode).

   ln file1 file2 → creates a hard link named file2 pointing to file1

2. Soft Link (Symbolic link): Shortcut pointing to the file path.

   ln -s file1 link1 → creates a symbolic link named link1 pointing to file1

LVM (Logical Volume Manager):

1. pvcreate /dev/sdb
2. vgcreate myvg /dev/sdb
3. lvcreate -L 5G -n mylv myvg
4. mkfs.ext4 /dev/myvg/mylv
5. mount /dev/myvg/mylv /mnt

📦Package Installations (using pip, a Python package installer):

1. pip list → list installed packages
2. pip show packagename → show details of a package
3. pip install packagename → install a Python package.
4. pip uninstall packagename → uninstall a Python package.
5. pip freeze > requirements.txt → freeze the installed packages into a requirements file.
6. pip install -r requirements.txt → install packages from a requirements file.

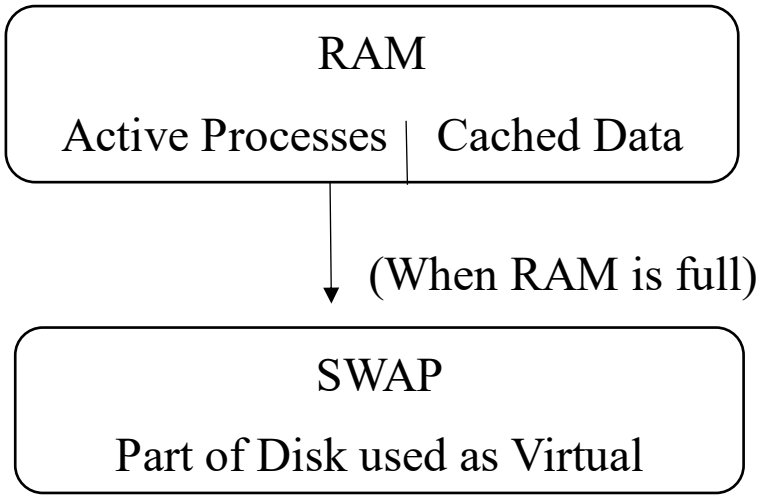
🔍Search and Find:

1. locate filename → find a file by its name. The database updated by updatedb command.
2. whereis programname → locate the binary, source, and manual page files for a command.
3. which commandname → shows the full path of (shell) commands
4. updatedb → mention that it may require sudo and is part of mlocate package

📊 System Monitoring and Performance:

1. iostat → reports Central Processing Unit (CPU) statistics and input/output statistics for devices, partitions, and network filesystems.
2. vmstat → reports information about processes, memory, paging, block IO, traps, disks, and CPU activity.
3. htop → an interactive process viewer for Unix systems. It's a more user-friendly alternative to top.
4. sar (system activity report, from sysstat).
5. dstat (combines vmstat, iostat, netstat).
6. htop, dstat, sar → may need installation: sudo apt install htop dstat sysstat

Memory Management



- **RAM** – volatile memory used by running processes
- **Swap Memory** – disk space used when RAM is full
    - swapon -s → check swap usage
    - free -h → RAM and swap info

- sudo swapon /swapfile → enable swap file
- sudo fallocate -l 2G /swapfile → create swap file
- mkswap /swapfile → format swap
- swapon /swapfile → activate swap

Others (mostly used in scripts):

1. command1 ; command2 → runs both regardless of success
2. command1 && command2 → second runs only if first succeeds
3. command1 || command2 → second runs only if gropunames fails
4. command & → run command in background
5. yes > /dev/null & → use this command to push a system to its limit.

⚠ :(){ :|:& };: → a fork bomb – handle with care. Do not run this command on a production system.

This command recursively spawns processes and can crash the system. Never run it outside of a controlled test environment.

📌 **Final Notes to Add at End**

1. Always use man <command> for documentation.
2. Use --help flag (e.g., ls --help).
3. Distros differ: apt (Debian/Ubuntu), yum/dnf (RHEL/CentOS), zypper (SUSE).
4. uname -r → show kernel version

5. lsb_release -a → show distro info (Debian/Ubuntu)
6. hostnamectl → show system hostname and OS info

□ Advanced Linux Admin Tips

🔄 Backup & Restore

- rsync -avh /source /destination → sync files/directories
- tar -czvf backup.tar.gz /path/to/dir → compress backup
- dd if=/dev/sda of=/backup.img → create disk image
- Tools: Timeshift, Deja Dup (GUI options)

🔐 Security Hardening

- fail2ban → block IPs with suspicious login attempts
- chkrootkit, rkhunter → scan for rootkits
- auditd → audit system events

🛡 SELinux & AppArmor

- SELinux: getenforce, setenforce
- AppArmor: aa-status, aa-enforce

📦 Containerization (Docker/Podman)

- docker run -it ubuntu bash → run container
- docker ps, docker images, docker exec → manage containers
- podman → daemonless alternative to Docker

⚙️ Performance Tuning

- sysctl → kernel parameter tuning
- ulimit → user-level resource limits
- nice, ionice → control CPU and I/O priority

🗂️ Filesystem Types

- Common: ext4, xfs, btrfs, zfs
- Mounting: mount -t ext4 /dev/sda1 /mnt

🌐 Network Troubleshooting Tools

- tcpdump → packet capture
- iftop, iptraf → live bandwidth usage
- ethtool → NIC diagnostics
- nc (netcat) → test ports and connections